PROCEEDINGS

55. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

13 - 17 September 2010

Crossing Borders within the ABC

Automation,

Biomedical Engineering and

Computer Science

Faculty of
Computer Science and Automation

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT
ILMENAU

Impressum
Published by

Publisher:	Rector of the Ilmenau University of Technology Univ.-Prof. Dr. rer. nat. habil. Dr. h. c. Prof. h. c. Peter Scharff
Editor:	Marketing Department (Phone: +49 3677 69-2520) Andrea Schneider (conferences@tu-ilmenau.de) Faculty of Computer Science and Automation (Phone: +49 3677 69-2860) Univ.-Prof. Dr.-Ing. habil. Jens Haueisen
Editorial Deadline:	20. August 2010
Implementation:	Ilmenau University of Technology Felix Böckelmann Philipp Schmidt

USB-Flash-Version.

Publishing House:	Verlag ISLE, Betriebsstätte des ISLE e.V. Werner-von-Siemens-Str. 16 98693 Ilmenau
Production:	CDA Datenträger Albrechts GmbH, 98529 Suhl/Albrechts
Order trough:	Marketing Department (+49 3677 69-2520) Andrea Schneider (conferences@tu-ilmenau.de)

ISBN: 978-3-938843-53-6 (USB-Flash Version)

Online-Version:

Publisher:	Universitätsbibliothek Ilmenau ilmedia Postfach 10 05 65 98684 Ilmenau

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=16739

CONTROL ALGORITHMS OF PLANAR MULTICOORDINATE POSITIONING SYSTEMS

V. Polyakovsky / S. Karpovich

1. ABSTRACT

The motion program building approach the goal of which is collateral control of several actuators bringing the affix the desired dynamical and accuracy properties, should be implemented in the way to avoid actuators collisions and to calculate the optimal trajectory based on set of control points. The algorithm which enables this is described in the paper. It is based on a geometrical representation of actuators movement and takes into consideration the priority system which is also dynamically built to furnish the system the maximal productivity.

2. INTRODUCTION

In computer-aided equipment of micro- and nanoelectronics items production multicoordinate systems of positioning configured out of several planar positioners placed on one stator which are meant for realization of technological and transport environment of production are rather perspective.

In such systems planar positioners should provide required joint motion on the fixed trajectories considering possible geometrical collisions. Such task is especially actual for computer-aided testers of multilayer printed-circuit boards in which technological operation of the control is carried out by probe manipulator mounted on 4 planar positioners which are placed in pairs on top and bottom stators relative to the control zone [1].

The basic problem in control algorithms of computer-aided testers of multilayer printed-circuit boards is in tracing the trajectories of motion of each of the positioners. It is desirable that it is optimum in speed, in fixed coordinate set of contact points with fixed pairwise combination of their bypass.

Therewith, in the process of generation of positioners programmed movements, it is necessary to take into account stator's geometrical areas, which positioners take when to provide joint movements without collisions.

3. CONTROL ALGORITHMS

3.1. Collision avoiding algorithm for linear low of motion

The geometrical collision avoiding model is presented on fig. 1. A number of planar linear stepper motors (LSM) denoted on fig. 1 as CP1, CP2, CP3 are moving on the stator across appropriate trajectories determining the areas of their movements (fig. 1).

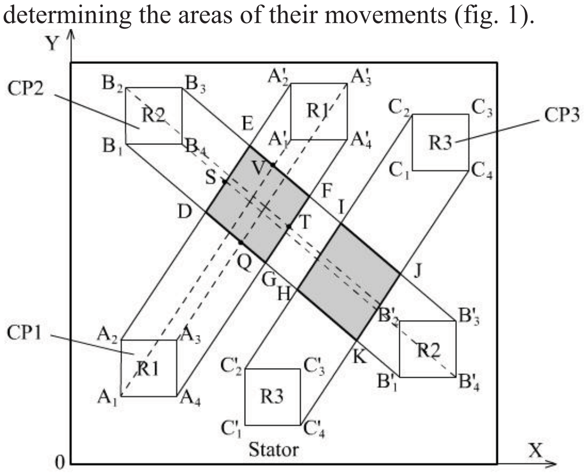

Fig. 1 Geometrical collision avoiding model

It is assumed that rectangular band (fig. 1) determines the area inside which all LSM are moving. The collision avoiding model is based on geometrical and kinematical analysis of LSM tail areas intersection. For example, for LSM CP1 and CP2 this is a tetragon DEFG (fig. 1).

Geometrical model analysis gives the following situations as the result:

- there is no collisions during LSM movement, as no tail areas intersection occur;
- collisions can occur during LSM movement, due to their tail areas intersection. This is the case when collision avoiding procedure will start and build a new movement program.
- special cases with critical zones which occur in some special configuration, when LSM movement is possible in special condition of impossible at all.

Collision avoiding algorithm is worked out for the case of LSM movement with constant speed across appropriate trajectories [1].

The parametric equations for any LSM position is as follows:

$$\begin{cases} x = x_0 + v_x t, \\ y = y_0 + v_y t, \end{cases} \quad (1)$$

where $v = \sqrt{v_x^2 + v_y^2}$ is LSM speed.

The equation for LSM linear trajectory is:

$$y = k \cdot x + b, \tag{2}$$

where $k = tg\varphi = \frac{v_x}{v_y}$.

The parameters of LSM trajectory are calculated as follows:

$$k = \frac{y_1 - y_0}{x_1 - x_0}, \tag{3}$$

$$b = y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0, \tag{4}$$

where x_0, y_0 – initial point coordinates; x_1, y_1– final point coordinates.

As it is seen from fig. 1, the area with collision possibility is the area with LSM tail are intersection (DEGF area for CP1 and CP2). On every step, the analysis of intersection zone is carried out, and the LSM entry into intersection zone is prohibited in case the other LSM is already inside this zone.

For example, for LSM CP1 the entry point into intersection zone is Q, exit point is V (fig. 1). The LSM CP2 enters intersection are in point S and exits the area in point T. The coordinates of these points are calculated according to the following equations:

$$\left[\begin{aligned} Xs &= \frac{b_{B_4B'_4} - b_{A_2A'_2}}{k_1 - k_2}, \\ Ys &= \frac{b_{B_4B'_4} - b_{A_2A'_2}}{k_1 - k_2} \cdot k_1 + b_{A_2A'_2}, \end{aligned}\right. \tag{5}$$

$$\left[\begin{aligned} X_T &= \frac{b_{B_2B'_2} - b_{A_4A'_4}}{k_1 - k_2}, \\ Y_T &= \frac{b_{B_2B'_2} - b_{A_4A'_4}}{k_1 - k_2} \cdot k_1 + b_{A_4A'_4}, \end{aligned}\right. \tag{6}$$

$$\left[\begin{aligned} X_Q &= \frac{b_{A_3A'_3} - b_{B_1B'_1}}{k_2 - k_1}, \\ Y_Q &= \frac{b_{A_3A'_3} - b_{B_1B'_1}}{k_2 - k_1} \cdot k_2 + b_{B_1B'_1}, \end{aligned}\right. \tag{7}$$

$$\left[\begin{aligned} X_V &= \frac{b_{A_1A'_1} - b_{B_3B'_3}}{k_2 - k_1}, \\ Y_V &= \frac{b_{A_1A'_1} - b_{B_3B'_3}}{k_2 - k_1} \cdot k_2 + b_{B_3B'_3}, \end{aligned}\right. \tag{8}$$

where *X, Y* are the coordinates of appropriate points (fig. 1);

$b_{A1A'1}$, $b_{A2A'2}$, $b_{A3A'3}$, $b_{A4A'4}$, $b_{B1B'1}$, $b_{B2B'2}$, $b_{B3B'3}$, $b_{B4B'4}$ are constant parameters of trajectory which are found from linear equations describing lines $A_1A'_1$, $A_2A'_2$, $A_3A'_3$, $A_4A'_4$, $B_1B'_1$, $B_2B'_2$, $B_3B'_3$, $B_4B'_4$ respectively.

k_1, k_2 are angle coefficients of appropriate trajectory.

When working out algorithm, the following priorities rule was taken into account: the priority of movement belongs to LSM which is the first to enter the intersection zone (fig. 1). The calculation of intersection zone entry times, the following equations were used:

$$t_{1\,in} = \frac{\sqrt{(X_Q - X_O)^2 + (Y_Q - Y_O)^2}}{V_1}, \tag{9}$$

$$t_{2\,in} = \frac{\sqrt{(X_S - X_M)^2 + (Y_S - Y_M)^2}}{V_2}, \tag{10}$$

where X, Y are the coordinates of appropriate points (fig. 1);

V_1, V_2 – speed of appropriate LSM.

The comparison of intersection zone entry times gives the number of LSM with higher priority, thus bringing it the movement with no stopping.

The calculation of stop time for another LSM which has the lower priority and therefore will wait for LSM with higher priority is as follows:

$$t_{1\,out} = \frac{\sqrt{(X_O - X_V)^2 + (Y_O - Y_V)^2}}{V_1}. \tag{11}$$

In case t_{2in}<t_{1out}, the LSM with lower priority is to stop for the time given by the following equation:

$$\Delta t = t_{out1} - t_{in2}. \tag{12}$$

On the basis of equations (1…12) collision avoiding algorithm was worked out.

3.2. Collision avoiding algorithm for linearly accelerated low of motion

Based on described mathematic model of collision avoiding algorithm for linear low of LSM motion we can obtain equations for linearly accelerated low of motion.

The parametric equations for any LSM position is as follows:

$$\begin{cases} x = x_0 + v_x t + \dfrac{a_x \cdot t^2}{2}, \\ y = y_0 + v_y t + \dfrac{a_y \cdot t^2}{2}, \end{cases} \tag{13}$$

where $\sqrt{a_x^2 + a_y^2} = a$ is LSM acceleration.

The equations to calculate intersection time entry are following:

$$t_{1\text{in}} = \frac{\sqrt{(X_Q - X_O)^2 + (Y_Q - Y_O)^2}}{V_1} + \frac{1}{2} t_{1a}, \quad (14)$$

$$t_{2\text{in}} = \frac{\sqrt{(X_S - X_M)^2 + (Y_S - Y_M)^2}}{V_2} + \frac{1}{2} t_{2a}, \quad (15)$$

where t_{1a} , t_{2a} - acceleration time of LSM.

The equation to calculate intersection time out is following:

$$t_{1\text{out}} = \frac{\sqrt{(X_O - X_V)^2 + (Y_O - Y_V)^2}}{V_1} + \frac{1}{2} t_{1a}. \quad (16)$$

Based on described mathematic model analytic collision avoiding algorithm is developed for multicoordinate position system. The sequence of operations of the developed algorithm is following:

1) tail areas calculation;
2) tail areas intersection calculation for all pairs of LSMs;
3) calculation of the coordinates of entry and exit points for each LSM based on equations (5…8);
4) calculation of intersection zone entry/exit time for all LSMs of the system using equations (14…16);
5) make a decision of existing or absence of collisions using equation (12);
6) decide on the actions to avoid collisions.

The block diagram of described algorithm is presented on fig. 2.

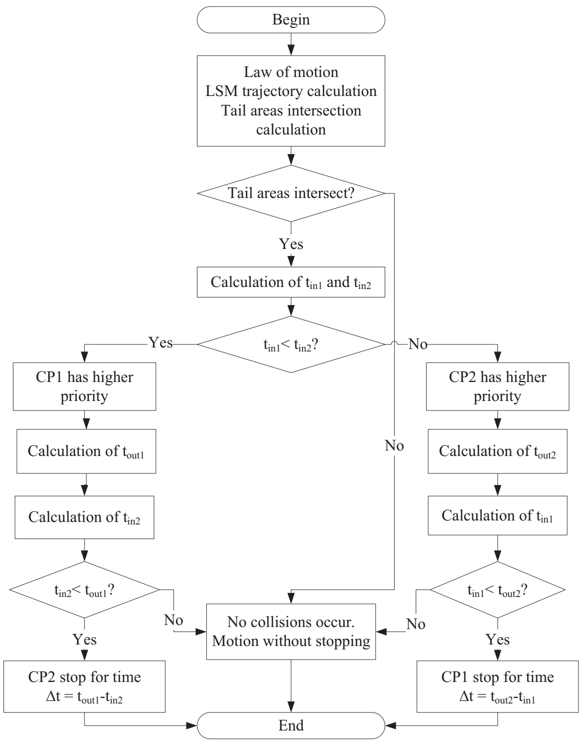

Fig. 2 Collision avoiding algorithm block diagram

As it is seen from fig. 2, the worked out algorithm enables collision-free motion building for the cases when either there is no collisions during LSM movement, as no tail areas intersection occur or for the case when collisions can occur during LSM movement, due to their tail areas intersection. The program does not take into account the special case when LSM movement is possible with special conditions.

3.3. Trajectory tracing algorithm

The task to trace trajectory of motion can be solved by various methods [2]:

- analytical method, through the analysis of point clouds by binary combinations of contact points;
- method of solution graph construction;
- method of optimum programming on possible set of solution-graphs.

Solution of the task of collisions at trajectories tracing is implemented in the form of computational procedure, aimed to account the constraint set which is providing the absence of mutual interpolations of positioners at their movements within one stator.

Based on the proposed comprehensive approach to tracing trajectories of planar positioners movements, it has been proposed a trajectories tracing algorithm for the control system of a multicoordinate positioning system. According to this algorithm on each subsequent movement of positioners from point to point it's being generated the law of movements in analytical or numerical type which is accepted for realization or rejected depending on the results of calculation based on the algorithm of analysis and collisions prevention. Consequently after successful performance of the tracing realized by the proposed algorithm, trajectories of planar positioners movements are being generated as well as movement laws optimum in speed, which realization in the planar positioner control systems excludes collisions probability.

4. THE INTERACTIVE VISUALIZATION AND MOTION DIAGRAM BUILDING

The collision avoiding algorithm presented on fig. 2 and described trajectory tracing algorithm were used as the basis for interactive simulation program implemented based on Adobe Flash technology.

The interface of implemented program is presented on fig. 3.

The program enables visualization and modeling of any configuration of multicoordinate positioning system based on LSMs. The result of modeling of multicoordinate positioning system with three LSMs A,B,C which are moving inside the tetragon working area (stator) is presented on the fig. 3.

The starting and finish point are set with the help of mouse. The movement speed of appropriate LSM is set in velocity window (fig. 3).

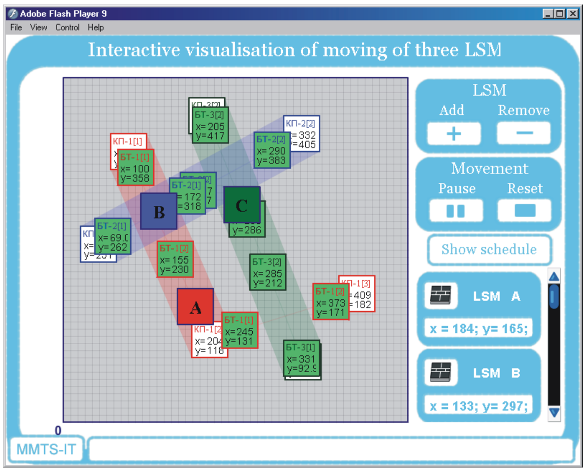

Fig. 3 Collision avoiding visualization program interface

After pressing "Start" button, the visualization starts. The program gives the possibility of all LSM motion diagram building (fig. 4).

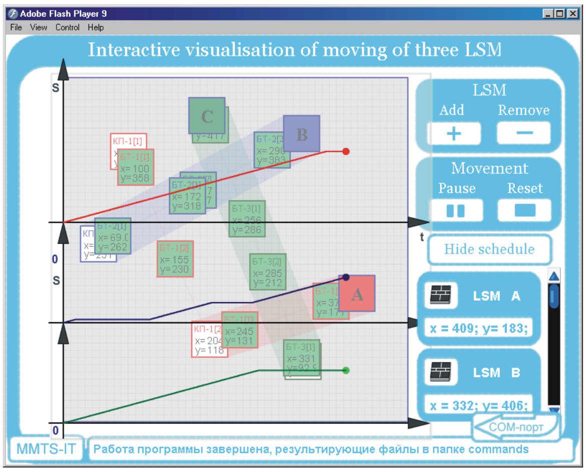

Fig. 4 LSM motion diagram building

According to movement program, the initial and final points of all LSM are set to fulfill the appropriate technological operation. As soon as calculations are performed, the result in the form of motion diagram can be exported for the purposes of LSM control system programming.

The problem of cooperative movement without collisions with possibility to trace trajectory is actual for a number of microelectronics technological equipment, for example, for probing equipment, where automatical circuit board testing is possible with four and more probes which are positioned on planar LSM of upper and lower stator [3].

The program helps in solving the problem of collision-free LSM movement and helps motion diagram building for the purposes of control system programming.

5. CONCLUSION

1. Motion program building algorithm which enables collision-free movement of several actuators along the stator is described in the paper. It is based on a geometrical representation of actuators movement and takes into consideration the priority system which is also dynamically built to furnish the system the maximal productivity.
2. Algorithm of tracing trajectory base of the set of the trajectory points is presented in the paper. This algorithm uses concept of greedy algorithms which are used to solve transport problem.
3. On the basis of worked out algorithms, the interactive visualization program was implemented which enables to simulate multicoordinate positioning systems and build motion diagram for all LSM as well, therefore brining the possibility of control systems' programming basing on simulation results.

6. REFERENCES

[1] Karpovich S., Rusetsky A., Liashuk Y., Matiushkov V. The Optimal Designing of Precision Coordinate Systems and Mechanisms of Electronic Technology Production Equipment.: Minsk, "Integral", 1999 (in Russian).

[2] Thorup M. Undirected Single-Source Shortest paths with positive integer weights in linear time / M. Thorup // Journal of the ACM, 46(3) - p 363-394, 1999.

[3] Karpovich S., Mezhinsky Y. Design Aspects of Flexible Manufacturing Systems with Linear Stepping Motors // Proceedings of 41th Internationales wissenschaftliches Kolloquium. Ilmenau, Germany, 1996. - P. 166-170.